Ghazaei G, Alameer A, Degenaar P, Morgan G, Nazarpour K. [An exploratory study on the use of convolutional neural networks for object grasp classification](#). In: *2nd IET International Conference on Intelligent Signal Processing 2015 (ISP)*. 2015, London, UK: Institution of Engineering and Technology.

Copyright:

© 2015 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works.

DOI link to article:

<http://doi.org/10.1049/cp.2015.1760>

Date deposited:

31/01/2018

An exploratory study on the use of convolutional neural networks for object grasp classification

G. Ghazaei^{1,†}, A. Alameer¹, P. Degenaar^{1,2}, G. Morgan³ and K. Nazarpour^{1,2,†}

¹*School of Electrical and Electronic Engineering, Newcastle University, Newcastle NE1 7RU, UK*

²*Institute of Neuroscience, Newcastle University, Newcastle NE2 4HH, UK*

³*School of Computing Science, Newcastle University, Newcastle NE1 7RU, UK*

[†]*Email for correspondence: {G.Ghazaei1, Kianoush.Nazarpour}@Newcastle.ac.uk*

Keywords: Artificial vision, Convolutional neural network (CNN), Hand prosthesis, Object recognition

Abstract

The loss of hand profoundly affects an individual's quality of life. Prosthetic hands can provide a route to functional rehabilitation by allowing the amputees to undertake their daily activities. However, the performance of current artificial hands falls well short of the dexterity that natural hands offer. The aim of this study is to test whether an intelligent vision system could be used to enhance the grip functionality of prosthetic hands. To this end, a convolutional neural network (CNN) deep learning architecture was implemented to classify the objects in the COIL100 database in four basic grasp groups: tripod, pinch, palmar and palmar with wrist rotation. Our preliminary, yet promising, results suggest that the additional machine vision system can provide prosthetic hands with the ability to detect object and propose the user an appropriate grasp.

1 Introduction

Computer vision has become an indispensable field in science and technology through providing human-like capability of seeing and visually sensing the world for a machine or a computer. There has been a significant focus on applications of computer vision in robotics and industrial applications. It has been however under-used in medical applications. One of these medical applications is prosthetic hand, which is the focus of this paper.

Prosthetic limbs provide a vital route to functional rehabilitation of amputees and people with congenital limb loss. Advanced prosthetic hands can improve dramatically amputees' quality of life by enabling them to do their activities of daily living and by helping them to return to their career and generate income. Despite considerable technical advances, the control of these commercial hands is limited to 1-dimension [1, 2]. In addition, the process of switching a prosthetic hand into an appropriate grip mode (e.g. pinch, tripod) to pick up different

objects can be cumbersome. Therefore, to become fully integrated into users' motor repertoire, the performance of hand prostheses must still improve greatly.

Conventionally, prosthetic hands are controlled with the electrical activity of muscles, the electromyogram (EMG) signal, recorded from the surface of the skin [2, 3]. Recently however alternative modalities have been introduced to replace the EMG. Skin movement analysis via accelerometry signals [4, 5] and computer vision [6–8] are two examples. Specifically in the case of the latter, by employing computer vision, object shapes can be analysed and an appropriate grasp can be determined.

There have been few studies that focused on utilising vision for hand prostheses. For example, in the work presented in [6, 7, 9], a dexterous hand was provided with vision and an autonomous controller. After triggering the hand and controlling its orientation by the user, a cognitive vision system (CVS) captured an image of the object. Then, the object's dimensions and distance to the target were estimated for the high level controller. The high level controller automatically proposes the best grasp type and size through rule-based reasoning. More recently, a semi-autonomous prosthetic hand control mechanism was developed in [8] with stereovision cameras and augmented reality (AR). Their proposed system could reduce the cognitive burden on user by using two sources of information, namely the EMG signals and artificial vision. Whether it is practical to ask users to wear AR glasses is debatable. In [10], deep learning for detecting robotic (not prosthetic) grasps was proposed to improve robotic grasp functionality.

The proposed project, however, follows a fundamentally different approach. We hypothesised that development of an automatic object grasp identification based on *abstract learning*, rather than estimating grasp on an object-by-object basis, could effectively enhance grip functionality of prosthetic hands. To this end, and as a proof of principle, we used convolutional neural network (CNN) [11] for our deep learning structure. Not only CNN has the ability to learn specific patterns, but also it is well suited to image classification tasks [12]. They are shown to perform well in a variety of challenging object recognition tasks [13]. However, other classification techniques,

Fig. 1: COIL100 dataset of 100 different objects

such as many of the conventional algorithm [14] or those inspired by biology [15, 16] could have been used. Designing an efficient and robust architecture for the object recognition task, the best grip mode for the detected object can be proposed by the model.

2 Methods

2.1 Dataset

The COIL100 dataset [17] was used for the recognition task. This dataset consists of 7200 images including 100 categories of objects. It is a dataset entirely including objects; most of which being household objects. There are 72 different poses per object, which are obtained by taking photos of each item against a black background at each 5° rotation through a turntable that covers 360° . The images are size normalised and the images with the dimension of 32×32 have been used in the tests. Figure 1 represents the COIL100 dataset.

2.2 Convolutional neural networks (CNNs)

CNNs are biologically inspired from the visual cortex, in which cells are sensitive to small sub-regions of the visual field called receptive fields. CNNs employ three architectural ideas, which provide them with some extent of shift, scale and distortion invariance: local connectivity, parameter sharing and pooling or sub-sampling. There are mainly two types of layers in a CNN: convolutional layers and sub-sampling or pooling layers. The former have different number of filters for extracting desired distinctive features, such as edges. These filters are applied across the image and the result is the feature maps that have the same number as the filters. Each feature map extracts

the same feature regardless of its location in the image. Subsequently, a non-linear down-sampling is applied to the feature maps in the next layer (sub-sampling or pooling). Figure 2 illustrates the implemented CNN in this project.

Having m^l images of size $r \times r$, k^l features are learned in each layer (l is the number of layers that is 0 for input layer) through small $c \times c$ ($c < r$) filters (kernels) according to

$$\mathbf{X}_{ij}^l = f(\mathbf{Z}_{ij}^l), \quad \mathbf{Z}_{ij}^l = (\mathbf{K}_j^l * \mathbf{X}_{ij}^{l-1} + \mathbf{b}_j^l) \quad (1)$$

$$i = 1, 2, \dots, m^l, \quad j = 1, 2, \dots, k^l$$

where, m^l is the number of input maps in each layer and k^l is the number of kernels in each layer. f is the activation function applied in an element-by-element manner, and $\mathbf{K}_j^l$ and $\mathbf{b}_j^l$ denote the j -th kernels and biases in l -th layer. $\mathbf{X}_{ij}^l$ is $(r - c + 1) \times (r - c + 1)$ matrix of convolved features (feature maps) of i -th input map and j -th filter in l -th layer, while $\mathbf{X}_i^{l-1}$ is the i -th input map of the l -th layer. “*” sign refers to a “valid” convolution meaning that convolution is performed inside the image borders.

The next layer in CNN is normally pooling, in which a region size is selected, e.g. $s \times s$, where $(r - c + 1)/s$ is an integer. The convolved features are partitioned to $s \times s$ sub-regions. In each sub-region, typically, the maximum or average is selected. The result can be used for classification through a fully connected network or followed by several other layers of convolution and pooling and then classified. We used Rectified Linear Unit (ReLU) [18] and max-pooling (used in [13, 19, 20]) as the activation and pooling functions respectively (Equations 2, 3).

$$a(t) = \max(0, t). \quad (2)$$

$$s_j = \max_{i \in R_j} c_i \quad (3)$$

where $a(\cdot)$ indicates the activation function and c_i is each element of the pooled feature map matrix in the pooling region R_i . The filters are then learned through back propagation and updating via stochastic gradient descent (SGD).

2.3 Grasp identification

In order to implement a CNN for grasp identification, the objects were manually organized based on their appropriate grasp type. In grasp recognition test, four common grip modes were chosen: Palmar, pinch, tripod and palmar with wrist rotation. In this way, rather than classifying thousands of examples in 100 classes, they are classified in 4 groups. The training set included 90% of the randomly arranged poses of each object in each grasp class (65 out of 72 images per object). The remaining 10% of the views for each object were given to the test set (3).

2.4 Cross-Validation

In order to achieve the variability of results based on weight initialisation and view selection, 10 different random sets of poses were selected. Then, for each set, 10 different random weight initialisations were examined.

Fig. 2: A CNN including convolution, pooling, convolution, and Softmax regression layers.

Fig. 3: The procedure of selecting train and test samples for a sample object. This process happens 10 times randomly for cross-validation.

3 Results

As an initial step, the performance of classifying the objects for the WOC test by using Softmax classifier [21] only was examined. Having four grasps as the outputs, the average accuracy for Softmax was $88.6 \pm 6.21\%$ [10-fold cross-validated across both weight initialisation and pose selection]. Cross-validated results for one-layer CNN indicates 94% grasp recognition average accuracy. Palmar grasp achieved the best accuracy among all the grasp types. That could be due to the small variety of objects in that group and the fact that most of them are

symmetric. The palmar grasp with wrist rotation is the second best class on average accuracy, probably because of the high number of training examples in this group. The large number of samples in the database led the algorithm to have a better chance to learn, even if there is a large variety of objects in that grasp group.

Finally, tripod and pinch group both have similar performance. The main reason could be the difference in different views of the same object; e.g. a toy car seems rectangular from 0° view, while it seems to have a square-like shape from 90° angle (Figure 3). Having more training data for these two groups could be helpful in improving the average accuracy. The average accuracy for two-layer CNN is 97.32% and it is above 96.6% ($\pm 0.8\% - \pm 2.76\%$ across pose selection and $\pm 0.17\% - \pm 0.4\%$ across weight initialisation) for each grasp. Both one and two-layer CNNs, presented better average accuracy than Softmax. Figure 4 indicates the average performance for the three architectures.

Considering deep architectures, two-layer CNN outperformed the one-layer CNN. With a two layer CNN, the classification accuracy in all groups was comparable. Yet, the group ranking, in terms of accuracy, remained the same. These improvements could be because a two-layer CNN, can learn more abstract features of the input data thanks to its additional free parameters. Having higher level of abstraction and neglecting the details could provide a better pose-invariance. It is worth mentioning that when the second pooling filter dimension was the same size (2) as that of the previous layer, the performance dropped to approximately 90%. This observation indicates that this pooling layer was excessive and sub-sampling eliminated helpful information provided in feature maps. Consequently, no pooling layer was added to the two-layer CNN.

Fig. 4: The cross-validated average accuracy and STD across pose selection and weight initialisation.

Larger standard deviation across view selection than that across weight initialisation implies that CNN is less robust to pose selection than weight initialisation. This seems plausible, since pose selection can lead to testing a view, which is unfamiliar to the trained algorithm and makes the recognition task more challenging.

4 Discussion

The shallow learning method examined in this project was Softmax regression, which was later used as a classifier in the deeper network (CNN) as well. The achieved results with Softmax higher than chance level. A reason for this shallow classifier is successful could be that the size of input images in the COIL100 (32×32) is small. Trying the Softmax on larger image sizes would not bring about as good result as deeper network used. As the real images are not as small as the images in the COIL100, a more complex way of learning is required. More importantly, even with the small size of images and few number of objects per class for the BOC test, both one-layer and two-layer CNNs show superior performance compared to Softmax.

Finding the best number of layers is a bit of challenging task. Adding layers to a neural network usually leads to learning more abstract features. That is, by having more layers in a deep network, filters become more complex and improve from simple details, e.g. edges, to more specific features. Interestingly, a deeper network does not essentially lead to better learning due to several reasons. One reason could be that there are more parameters to be tuned in a deeper network, so finding the best parameters to have the best accuracy is more challenging. Having a deeper network, more data also should be

prepared for training to avoid over-fitting. In our tests, the two-layer CNN model achieved better results than the one-layer CNN model. This could be because compared to object identification, grasp recognition should be able to estimate the appropriate grasp other than the pose of the same object. A higher level of abstraction, which is provided by two-layer CNN leads to neglecting details and offers better generalisation for the test. This abstraction can provide superior performance in grasp recognition.

5 Conclusion

We proposed a simple deep learning structure to classify objects in a standard and public dataset into four groups according to the grasp type required for handling them. We showed that a deep CNN structure outperforms shallow classifier in classifying unseen aspects of objects. Whether CNN can generalise to identify the grasp types for the objects that have not been seen at all during training is the subject of our future work. Once the CNN is learned the computational cost of implementing it in hardware, e.g. a prosthetic hand, is very small.

Acknowledgements

The work of G Ghazaei was supported by a DTA from Newcastle University. A. Alameer has a PhD scholarship from the HCED, Iraqi Government. The work of K. Nazarpour is supported by EPSRC, UK (grant reference numbers: EP/M025977/1 and EP/M025594/1).

References

- [1] K. Nazarpour, C. Cipriani, D. Farina, and T. Kuiken, "Advances in control of multi-functional powered upper-limb prostheses," *IEEE Trans. Neural Sys. Rehab. Eng.*, vol. 22, no. 4, pp. 711–715, 2014.
- [2] D. Farina, N. Jiang, H. Rehbaum, A. Holobar, B. Graimann, H. Dietl, and O. Aszmann, "The extraction of neural information from the surface EMG for the control of upper-limb prostheses: Emerging avenues and challenges," *IEEE Trans. Neural Sys. Rehab. Eng.*, vol. 22, no. 4, pp. 798–809, 2014.
- [3] T. Pistohl, C. Cipriani, A. Jackson, and K. Nazarpour, "Abstract and proportional myoelectric control for cursor and prosthetic hand," *Ann. Biomed Eng.*, vol. 41, no. 12, pp. 2687–2698, 2013.
- [4] M. Atzori, A. Gijsberts, C. Castellini, B. Caputo, A.-G. M. Hager, S. Elsig, G. Giatsidis, F. Bassetto, and H. Müller, "Electromyography data for non-invasive naturally-controlled robotic hand prostheses," *Sci. Data*, vol. 1, p. 140053, 2014.
- [5] A. Krasoulis, S. Vijayakumar, and K. Nazarpour, "Evaluation of regression methods for the continuous decoding of finger movement from surface EMG and accelerometry," in *IEEE Neural Engineering, International Conference*, 2015.
- [6] S. Došen and D. B. Popović, "Transradial prosthesis: artificial vision for control of prehension," *Artificial organs*, vol. 35, no. 1, pp. 37–48, 2011.
- [7] S. Došen, C. Cipriani, M. Kostić, M. Controzzi, M. C. Carrozza, and D. B. Popović, "Cognitive vision system for control of dexterous prosthetic hands: experimental evaluation," *Journal of Neuroengineering and Rehabilitation*, vol. 7, no. 1, p. 42, 2010.
- [8] M. Markovic, S. Dosen, C. Cipriani, D. Popović, and D. Farina, "Stereovision and augmented reality for closed-loop control of grasping in hand prostheses," *Journal of Neural Engineering*, vol. 11, no. 4, p. 046001, 2014.
- [9] D. Klisić, M. Kostić, S. Došen, and D. B. Popović, "Control of prehension for the transradial prosthesis: natural-like image recognition system," *Journal of Automatic Control*, vol. 19, no. 1, pp. 27–31, 2009.
- [10] I. Lenz, H. Lee, and A. Saxena, "Deep learning for detecting robotic grasps," *The International Journal of Robotics Research*, vol. 34, no. 4-5, pp. 705–724, 2015.
- [11] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.
- [12] L. Deng, "A tutorial survey of architectures, algorithms, and applications for deep learning," *APSIPA Transactions on Signal and Information Processing*, vol. 3, p. e2, 2014.
- [13] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, pp. 1097–1105, 2012.
- [14] D. G. Lowe, "Object recognition from local scale-invariant features," in *The proceedings of the seventh IEEE international conference on Computer vision, 1999.*, vol. 2, pp. 1150–1157, Ieee, 1999.
- [15] T. Serre, L. Wolf, S. Bileschi, M. Riesenhuber, and T. Poggio, "Robust object recognition with cortex-like mechanisms," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 3, pp. 411–426, 2007.
- [16] A. Alameer, G. Ghazaei, P. Degenaar, and K. Nazarpour, "An elastic net-regularized h-max model of visual processing," in *Proc. 2nd IET Intelligent Signal Processing, International Conference*, London, UK, Dec. 2015.
- [17] S. Nayar, S. A. Nene, and H. Murase, "Columbia object image library (COIL 100)," *Department of Comp. Science, Columbia University, Tech. Rep. CUCS-006-96*, 1996.
- [18] X. Glorot, A. Bordes, and Y. Bengio, "Deep sparse rectifier networks," in *Proceedings of the 14th International Conference on Artificial Intelligence and Statistics. JMLR W&CP Volume*, vol. 15, pp. 315–323, 2011.
- [19] D. Scherer, A. Müller, and S. Behnke, "Evaluation of pooling operations in convolutional architectures for object recognition," in *Artificial Neural Networks–ICANN 2010*, pp. 92–101, Springer, 2010.
- [20] K. Jarrett, K. Kavukcuoglu, M. Ranzato, and Y. LeCun, "What is the best multi-stage architecture for object recognition?," in *2009 IEEE 12th International Conference on Computer Vision*, pp. 2146–2153, IEEE, 2009.
- [21] "Softmax regression toolbox." <http://ufldl.stanford.edu/tutorial/supervised/SoftmaxRegression/>,. Available online, Accessed: 29-01-2015.